

VIRTUAL NAVIGATION

Patent number: JP11250107

Publication date: 1999-09-17

Inventor: YOUNG JEFFREY E

Applicant: ADOBE SYSTEMS INC

Classification:

- international: G06F17/30; G06F17/30; (IPC1-7): G06F17/30;
G06F17/21; G06F17/27

- european: G06F17/30G4

Application number: JP19980363460 19981221

Priority number(s): US19970995313 19971222

Also published as:



EP0924629 (A2)

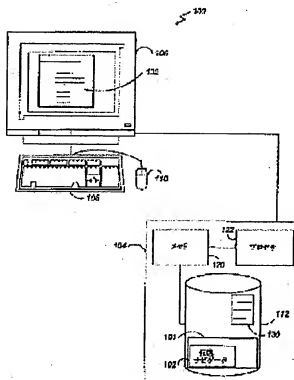
US6006236 (A1)

EP0924629 (A3)

Report a data error here

Abstract of JP11250107

PROBLEM TO BE SOLVED: To improve a method for identifying a link in an electronic document by traversing a data structure through the use of a base link and recognizing a characteristic shared by components so as to generate a virtual link between components within the data structure. **SOLUTION:** An electronic document publishing system 101 uses a base link for identifying the correlation of all the components in a hierarchical structure. The system 101 searches a specific component within a data structure by using a virtual navigator 102. The system 101 gives the navigator 102 to a component of each type requiring to be discriminated, namely identified. The navigator 102 uses the base link of hierarchical data structure or a viral link given by some other virtual navigator and recognizes a common characteristic used in common by a pair of components to recognize a pair of the components.



Data supplied from the esp@cenet database - Worldwide

【特許請求の範囲】

【請求項1】 複数部のコンポーネント及びコンポーネント間の構造的関係を指定するベースリンクを具備するデータ構造として表わされている電子ファイルにおけるリンクを識別するコンピュータによって実現される方法において、

複数部のベースリンクを使用してデータ構造をトラバースし、

第一コンポーネントと第二コンポーネントとによって共有されている特性を認識することによって前記データ構造における第一コンポーネントと第二コンポーネントとの間の仮想リンクを発生することを特徴とする方法。

【請求項2】 請求項1において、前記仮想リンクがランダムにおいて必要とされる場合に識別されることを特徴とする方法。

【請求項3】 請求項1において、更に、前記データ構造のトラバースを完了する前に第二コンポーネントを使用して構造を実行することを特徴とする方法。

【請求項4】 請求項1において、更に、コンポーネント間の仮想リンクを識別するために逐次的に実行する複数部のトラバースルーチンを与えることを特徴とする方法。

【請求項5】 請求項1において、前記第二コンポーネントがコンポーネントクラスから特徴を受継ぎ、且つトラバースルーチンが前記第二コンポーネントが見つかるまで前記コンポーネントクラスのメンバーを認識することによって前記第二コンポーネントを認識することを特徴とする方法。

【請求項6】 請求項5において、前記データ構造が階層的データ構造であり且つ前記トラバースルーチンがファミリー、次の及び前の構造的関係によってトラバース経路を特定することを特徴とする方法。

【請求項7】 請求項3において、前記電子ファイルが電子文書であることを特徴とする方法。

【請求項8】 請求項7において、前記第二コンポーネントに関して実行される機能が番号付き構造であることを特徴とする方法。

【請求項9】 請求項7において、前記第二コンポーネントに関して実行される機能がテキストを発生することを特徴とする方法。

【請求項10】 請求項7において、前記第二コンポーネントに関して実行される機能がテキストストリングを抽出することを特徴とする方法。

【請求項11】 請求項7において、前記トラバースルーチンがコンポーネント間の複数部の仮想リンクを識別することを特徴とする方法。

【請求項12】 請求項11において、前記データ構造が階層的データ構造であり、前記仮想リンクが前記階層的データ構造において階層的なサブセットのコンポーネントを表わし、前記トラバースルーチンがファミリー

一、次の又は前の構造的関係によってトラバース経路を識別し、且つ前記トラバースルーチンがデータタイプに従ってコンポーネントを特定することを特徴とする方法。

【請求項13】 ランダムで電子ファイル内のリンクを識別するためのコンピュータによって実現される方法において、

複数部のコンポーネント及び前記コンポーネント間の構造的関係を指定する複数部のベースリンクを具備する階層的データ構造として電子ファイルを生じ、

前記ベースリンクを使用する複数部のトラバースルーチンを使用して前記階層的データ構造をトラバースし、他のトラバースルーチンクラスから特徴を受継ぐクラスとして前記トラバースルーチンを確定し、

前記コンポーネントによって共有されている特性を認識することによって前記階層的データ構造における複数部のコンポーネント間の複数部のリンクを各トラバースルーチンによって識別させ、

前記コンポーネントが識別された時に各識別されたコンポーネントを使用して構造を実行することを特徴とする方法。

【請求項14】 複数部のコンポーネント及び前記コンポーネント間の構造的関係を指定する複数部のベースリンクを具備するデータ構造として提供されている電子ファイルに関して動作するコンピュータプログラムであって、コンピュータによって読取り可能な媒体上に存在しているコンピュータプログラムにおいて、コンピュータをして、

少なくとも1個のトラバースルーチンを供給させ、前記トラバースルーチンが前記ベースリンクを使用してデータ構造をトラバースすることによって前記データ構造における第一コンポーネント及び第二コンポーネントの間のリンクを識別する、上記命令を有することを特徴とするコンピュータプログラム。

【請求項15】 請求項14において、前記第二コンポーネントがあるクラスのコンポーネントから特性を受継ぎ且つ前記トラバースルーチンが前記クラスのコンポーネントのメンバーを認識することによって前記リンクを識別することを特徴とするコンピュータプログラム。

【請求項16】 請求項14において、更に、コンピュータをして、

前記第二コンポーネントが識別された時に前記第二コンポーネントを使用して構造を実行させる、上記命令を有することを特徴とするコンピュータプログラム。

【請求項17】 請求項14において、前記電子ファイルが電子文書であることを特徴とするコンピュータプログラム。

【請求項18】 請求項17において、更に、コンピュータをして、

前記データ構造内の複数部のコンポーネント間の複数部

のリンクを発生させる。上記命を有することを特徴とするコンピュータプログラム。

【請求項19】 請求項17において、更に、コンピュータをして、

前記データのトラバースが完了する前に前記第二コンポーネント及びその他のリンクされたコンポーネントを使用して機能を実行させる、上記命を有することを特徴とするコンピュータプログラム。

【請求項20】 請求項19において、前記実行される機能が番号付き直線構造であることを特徴とするコンピュータプログラム。

【請求項21】 請求項19において、前記実行される機能がテキストを発生することを特徴とするコンピュータプログラム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明は、電子文書におけるコンポーネント間構成要素を識別する技術に関するものである。

【0002】

【従来の技術】 電子文書は、典型的に、例えばテキスト、グラフィック、テーブルなどの情報内容と、その情報内容をどの様にして表示するかを指示するフォーマット内容とを有している。文書処理システムとデスクトップ出版システムとを包含する文書出版システムは、階層的データ構造として電子文書を格納する。この様な構造は、情報内容とフォーマット内容とを関係付けられた配列で互いに識別されているノードとして表わす。

【0003】 システムがデータ構造をトラバースして、該構造に関するデータを取扱し且つそのデータを使用して操作を実行する。階層的構造をトラバースするために、該システムは一つのノードから別のノードへ組のリンクに依拠する。

【0004】 ノード間のリンクは、時々、ファミリー関係によって区別される。階層的構造において別のノードへ取付けられており且つその上方のノードは後者のノードのペアレント即ち親として呼ばれる。階層的構造において別のノードへ取付けられており且つその下側にあるノードは後者のノードのチャイルド即ち子として呼ばれる。同一の親を有するノードはシブリングズ即ち子孫として呼ばれる。ファミリーリンクによってノード間関係を決定することによって、システムは、次のリンク及び前のリンクとが可能な。次のリンク及び前のリンクは、ファミリー関係を無視し且つ文書内のノードの相対的な位置を取扱う。

【0005】 ファミリーリンク及び次のリンク及び前のリンクを「ベースリンク」と呼称する。ベースリンクは、構造内の全てのノードを接続し且つその構造の階層を決定する。システムは構造をトラバースし且つ構造の

組織を発見するためにベースリンクを使用する。構造の組織は、あるタイプの操作に対する処理の順序を決定する。例えば、スペルチェッカーは、文章の始めから終りまで電子文書内の各ワードを検査するためにベースリンクを使用することが可能である。構造の組織は、X、Yのノードが他のノードと等価特性を共有するかを決定する。例えば、あるノードは、子孫のノードによって定義がれ且つ流線化されたパラグラフ特性を決定することが可能である。

【0006】 階層的データ構造で全てのノードを接続する一組のベースリンク以外に、システムは階層的データ構造の同一の又は異なる分岐におけるノードを接続するために幾つかの組の直接的リンクを有することが可能である。直接的リンクは、ある一組の条件下において互いに影響を及ぼす場合のあるノードを差し出す。例えば、著者が文書内に番号付けしたセクションのヘッディングを挿入した場合には、システムは番号付けしたセクションヘッディングノード間の組の直接的リンクを使用し且つ全てのその他のセクションヘッディングを差し出す且つ番号の付直しをすることが可能である。直接的リンクは、X、その他の場合においても有用であり。例えば、詳細なアウトラインのコンポーネントを識別すること、簡単なアウトラインのコンポーネントを識別すること、全てのインデックスマークを差し出すこと、且つ全ての著者目録引用を差し出すことに有用である。

【0007】

【発明が解決しようとする課題】 本発明は、以上の点に起因せられたものであって、上述した如き従来の技術の欠点を解消し、電子文書におけるリンクを識別する改良した方法及び装置を提供することを目的とする。

【0008】

【課題を解決するための手段】 一つの側面においては、本発明はコンポーネント（構成要素）とベースリンクとを具備するデータ構造として表わされている電子文書内のリンクを識別するコンピュータによって実現される方法を提供している。ベースリンクはコンポーネント間の構造的問題を決定即ち定義する。本発明方法は、ベースリンクを使用してデータ構造をトラバースし且つ該コンポーネントによって利用されている特性を認識することによってデータ構造内のコンポーネント間の位置リンクを発生する。

【0009】 位置リンクは、ランタイム即ち実行時間において必要とされる場合に識別される。例えば番号付け直し機能又はテキストを発生する機能などのある問題を、仮想的に別のコンポーネントへリンクされている各コンポーネントを使用して実行することが可能である。

【0010】 多数個のトラバースルーチンが逐次的に実行して、コンポーネント間の仮想リンクを識別することが可能である。データ構造は階層的なものとすることが可能であり且つトラバースルーチンによって使用される

トバース経路は、ファミリー、次の及び前の構造的關係によって表わすことが可能である。

【0011】本発明の効果としては、以下に記載するようである。本発明は、単に一部のベースリンクを必要とするに過ぎない。コンポーネント間のその他の全てのリンク（例えば、直接的なリンク）を除去することは、状態が変更される場合に他のリンクを再生させる必要性を排除している。更に、複製組のリンクが格納されるものでないからメモリ条件が緩和されている。

【0012】図2の矢印の形態図1を参照すると、電子文書出版システム101をサポートするに達したコンピュータプラットフォーム100が示されている。電子文書出版システム101は、ディスク上又はメモリ内に1個又はそれ以上の仮想ナビゲータ102を有している。コンピュータプラットフォーム100は、デジタルコンピュータ104、ディスプレイ106、キーボード108、マウス又はその他のポインティング装置110、大容量記憶装置112（例えば、ハードディスクドライブ、MOディスクドライブ、又はフロッピーディスクドライブ）を有している。コンピュータ104はメモリ120、プロセッサ122、例えばメモリバス及び周辺バス（不図示）などのその他の通常のコンポーネント即ち構成要素を有している。電子文書130は、ハードディスク又は例えばディスクなどのその他のコンピュータによって取り扱可能な媒体上に格納されている情報を有している。電子文書130の人間が閲覧可能なバージョンはコンピュータディスプレイ106上と、又はコンピュータプログラムによって電子文書に関する操作によって得られるハードコピープリントアウト上で観察することが可能である。

【0013】図2を参照すると、階層的なデータ構造200として組織化されている一群のコンポーネント201-206を示してある。データ構造200は電子文書を表わしている。これらのコンポーネントは、セクションヘッディング、パラグラフ、リスト項目などとして可能である。例えば、コンポーネント202及びコンポーネント205はパラグラフとすることが可能であり、コンポーネント203及び206は図2はインデックスメントとすることが可能である。

【0014】電子文書出版システム101は、この階層的な構造における全てのコンポーネントの相互関係を識別するためにベースリンクを使用する。図2におけるノード201-206の間の関係は250-256はデータ構造200のファミリーリンク及び次のリンク及び前のリンクを示している。ファミリーリンク及び次の及び前のコンポーネントリンクは、各コンポーネントと共に属性/値対として特定し且つ格納することが可能である。例えば、属性としては親リンク又は子供リンクとすることが可能である。

が可変であり且つ値は親ノード又は子供ノードへのポイントとすることが可能である。

【0015】例えば直接的なリンクなどの付加的なリンクを格納し且つ維持する代わりに、システム101は仮想ナビゲータ102（図1）を使用して、データ構造内の特定のコンポーネントを提出する。仮想ナビゲータはソフトウェアルーチンである。その名前が暗示するように、仮想ナビゲータは、ベースリンクを介してデータ構造をナビゲースすることによってコンポーネント間の見出しの経路を識別する。

【0016】図3Aには図2のコンポーネント203と、インデックスコンポーネント204と、図2のコンポーネント206との間に見出しの経路357及び見出しの経路358が示されている。図2のコンポーネント203及び206はインデックスコンポーネント204は、それらが、例えばパラグラフなどの別のコンポーネントへアンカーされており且つ両方共あるタイプのアンカーコンポーネントであるという特性を共有している。アンカー仮想ナビゲータが、ベースリンク255を使用することによって図2のコンポーネント203とインデックスコンポーネント204との間に仮想リンク357を発生し、且つベースリンク254、ベースリンク252、ベースリンク256を使用することによってインデックスコンポーネント204と図2のコンポーネント206との間に仮想リンク358を発生する。図3Bには仮想リンク359が示されている。仮想リンク359は仮想リンク357及び仮想リンク358から派生したものである。図2の仮想ナビゲータは、アンカー仮想ナビゲータが、図2の仮想リンク357及び仮想リンク358を使用して仮想リンク359を発生している。

【0017】電子文書出版システム101は、図3Bに示されることを必要とする各タイプのコンポーネントに対して仮想ナビゲータを有する。仮想ナビゲータ102の例は、全ての図2の図2を提出し出す仮想ナビゲータ。全ての順番付けされているリストを提出し出す順番付けリスト仮想ナビゲータ、全ての番号付けされているパラグラフを提出し出す番号付けパラグラフ仮想ナビゲータ、及び全てのパラグラフを提出し出すパラグラフ仮想ナビゲータなどがある。

【0018】オブジェクト指向環境においては、ベース仮想ナビゲータクラスは、それからその他の全ての仮想ナビゲータクラスが派生されて従ってその他の全ての仮想ナビゲータクラスがベース仮想ナビゲータクラスから特徴を受継ぐクラスである。仮想ナビゲータ102の各タイプは、それ自身のクラスによって定義され、且つ各仮想ナビゲータ102はそのクラスから例示された即ちインスタンスとして発生されたオブジェクトである。全ての仮想ナビゲータ102は祖先の仮想ナビゲータクラスに対して定義されている特徴を受継ぎ且つ使用することが可能である。

【0019】各仮想ナビゲータ102は簡易的データ構造のベースリンク又はその他の仮想ナビゲータによって与えられた仮想リンクを使用し且つ同一組のコンポーネントによって共有されている共通の特性を認識することによって一組のコンポーネントを識別する。仮想ナビゲータ102は、一組のコンポーネントを識別した後にコンポーネント属性上又はコンピュータメモリ内にデータ構造を維持するか又は格納することは必要ではない。一組のコンポーネントは同時に発見される別のコンポーネントは、仮想ナビゲータが一度に同一のコンポーネントをサーチする前に、そのコンポーネントが発見された後に特定の順序のために使用される。仮想ナビゲータは、番号がデータ構造200における1個又はそれ以上のコンポーネントを、何らかの順序で付加し、削除し、移動し、又は修正する場合に使用することが可能である。修正が他のコンポーネントが番号付けされている順序に影響を与える場合には、番号付直しルーチンがコール即ち呼出されて影響されるパラグラフに番号の付直しを行うことが可能である。そのルーチンは番号付けパラグラフ仮想ナビゲータ、即ち仮想ナビゲータ、又は両方を使用して、番号の付直しを必要とするコンポーネントを個別に再決定することが可能である。

【0020】一例として、電子文書内の既存のセクションヘッディング内に新たなセクションヘッディングが挿入される場合に仮想ナビゲータをコール即ち呼出することが可能である。従って、新たなセクションヘッディングがセクション2、0とセクション3、0の間に挿入される場合には、仮想ナビゲータがセクション3、0からその電子文書の終わりまでにおいて番号付けされている全てのセクションヘッディングを識別する。一つのセクションヘッディングが識別されると、仮想ナビゲータをコールしたルーチンのようなあるルーチンがヘッディングの番号の付直しを行う。セクション3、0はセクション4、0となり、セクション3、1はセクション4、1となるなどである。

【0021】仮想ナビゲータ102は直、次の子供、前の子供、第一の子供、最後の子供、且つ次の及び前のコンポーネントを探索するトラバース方法に基づくアルゴリズムを使用する。各仮想ナビゲータ102は特定のタイプのコンポーネントに対して探索されている多くとも1個のトラバースルーチンを実行し且つそのコンポーネントタイプに対するリンク条件を検討する。例えば、番号付けパラグラフ仮想ナビゲータは3個のトラバースルーチン、即ち、「GetParent」、「GetNext」、「GetPrev」を有しており、それらは番号付けされているパラグラフコンポーネントを識別する。パラグラフ仮想ナビゲータはパラグラフコンポーネントを認識するトラバースルーチン「GetParent」、「GetNext」、「GetPrev」、「GetNextChild」、「GetPrevChild」

「d」、「GetFirstChild」、「GetLastChild」を有している。

【0022】図4は仮想ナビゲータの使用を例示している。第一に、電子文書をベース構造リンクを有する簡易的データ構造200として格納する。電子文書出版システム101が特定のコンポーネントに関するタスクを実行することを必要とする場合には、それらのコンポーネントを識別するために仮想ナビゲータがコール即ち呼出される。識別されたコンポーネントの間のリンクは格納される。従って仮想ナビゲータはコンポーネントが識別される場合にコンポーネント間の仮想リンクを発生する（ステップ420）。仮想ナビゲータは、その他の仮想ナビゲータをコール即ち呼出することによって仮想リンクを発生させる。仮想ナビゲータは、特定の組のコンポーネントを識別するために、簡易的データ構造のベースリンクであって、単に一つのコンポーネントから別のコンポーネントへのポインタとすることの可能なベースリンクを使用する。

【0023】仮想リンクを発生させるために、仮想ナビゲータは、後に説明する特定の特性を持ったコンポーネントを識別する（ステップ460）。仮想ナビゲータをコールしたルーチンは識別したコンポーネントを使用して操作を実行することが可能である（ステップ470）。その操作を実行した後に、仮想ナビゲータは、特定された特性を持った別のコンポーネントをサーチするために再度コールされることが可能である。仮想ナビゲータをコールし且つ操作を実行することのサイクルは、コールするルーチンが全てのコンポーネントが識別されたことを判別するまで繰り返される。例えば、コールを行うルーチンは簡易的データ構造全体をトラバースすることを必要とするか、又は特定の分岐におけるコンポーネントを識別することが必要であるに過ぎない場合がある。

【0024】同様のリンク条件に起因して、仮想ナビゲータ102は、その他のタイプのコンポーネントを識別するその他の仮想ナビゲータをコールする。一体となつて、これらの仮想ナビゲータはベースリンクを介して簡易的データ構造全体をトラバースすることが可能である。例えば、順序付けリストコンポーネントは番号付けされたパラグラフコンポーネントがその親コンポーネントであることを必要とし、且つ番号付けされたパラグラフコンポーネントはパラグラフコンポーネントがその親であることを必要とする。この場合には、順序付けリスト仮想ナビゲータは番号付けパラグラフ仮想ナビゲータをコールし、且つ番号付けパラグラフ仮想ナビゲータはパラグラフ仮想ナビゲータをコールする。

【0025】図5は、GetNextトラバースルーチンを使用して順序付けリストコンポーネントを識別するために互いに相互作用を行う3個の仮想ナビゲータの概念的表示を示している。順序付けリストクラスは番号付

付パラグラフクラスから派生され且つ番号付けパラグラフクラスはパラグラフクラスから派生されるクラスである。順番付けリスト仮想ナビゲータは、順番付けリストコンポーネントを見つめるまで次の番号付けパラグラフを逐次獲得することによって次の順番付けリストコンポーネント(ステップ460')を獲得する(ステップ520)。次の番号付けパラグラフを獲得するために、番号付けパラグラフ仮想ナビゲータは、番号付けされたパラグラフを見つめるまで(ステップ530)、逐次的に次のパラグラフを獲得する。このカスケード探索は、全てのコンポーネントクラスが派生されるクラスにおける一つのコンポーネントを識別する仮想ナビゲータに到達するまで継続することが可能である。

【0026】図6を参照すると、順番付けリストであるコンポーネントを識別する順番付けリスト仮想ナビゲータのGetNextトランザクション460'の一例が示されている。順番付けリスト仮想ナビゲータのGetNextトランザクション460'は、その構造における次の順番付けされているパラグラフを獲得することによって開始する(ステップ521)。順番付けリスト仮想ナビゲータのGetNextトランザクション460'は、番号付けパラグラフ仮想ナビゲータのGetNextトランザクション520をコールする(ステップ521)。順番付けリスト仮想ナビゲータは、番号付けされているパラグラフがリターンされたか否か(ステップ522)及び番号付けされているパラグラフが順番付けリストコンポーネントであるか否か(ステップ524)をテストする。順番付けリストコンポーネントがリターンされていた場合には、順番付けリスト仮想ナビゲータがリターンし(ステップ526)且つコールしたルーチンはその順番付けリストコンポーネントを使用して所定の機能を実行することが可能である。例えば、その機能は、セクション番号をインクリメントすることが可能である。番号付けパラグラフがリターンされていたが、それは順番付けリストコンポーネントではなかった場合には、順番付けリスト仮想ナビゲータが継続して順番付けリストコンポーネントに対するサーチを行う。番号付けされたパラグラフがリターンされなかった場合には、その構造全体をトランザクションで順番付けリスト仮想ナビゲータはコールしたルーチンへリターンする(ステップ526)。

【0027】次の番号付けされているパラグラフを獲得することは同様の技術に従う。次の番号付けされているパラグラフを獲得するために、番号付けパラグラフ仮想ナビゲータのGetNextトランザクション520がパラグラフ仮想ナビゲータのGetNextトランザクション530をコールし(ステップ531)、パラグラフがリターンされたか否かをテストし(ステップ532)、且つパラグラフがリターンされていた場合には、そのパラグラフが番号付けされているパラグラフであるか否かをテストする(ステップ534)。そのパラ

グラフが番号付けされているパラグラフではなかった場合には、番号付けパラグラフ仮想ナビゲータのGetNextトランザクション520が、番号付けされているパラグラフがリターンされたか又はその番号付けパラグラフ仮想ナビゲータがその構造をトランザクションまで、ステップ531-534を繰り返して行う。

【0028】次のパラグラフを得るために、パラグラフ仮想ナビゲータは次のコンポーネントを獲得する。なぜならば、パラグラフはコンポーネントから派生されるからである。パラグラフ仮想ナビゲータのGetNextトランザクション530がコールされて、次のコンポーネントを獲得する。パラグラフ仮想ナビゲータのGetNextトランザクションがコンポーネント仮想ナビゲータのGetNextトランザクションをコールし(ステップ541)且つコンポーネントがリターンされたか否かをテストし(ステップ542)、且つそうである場合には、そのコンポーネントがパラグラフであるか否かをテストする(ステップ544)。そのコンポーネントがパラグラフでなかった場合には、パラグラフ仮想ナビゲータのGetNextトランザクションが、パラグラフがリターンされるか又はパラグラフ仮想ナビゲータがその構造をトランザクションまでステップ541-544を繰り返して行う。

【0029】以上、本発明の具体的な実施の形態について詳細に説明したが、本発明は、これらの実施の形態に限定されるべきではなく、本発明の技術的範囲を逸脱することなしに種々の変形が可能であることが加えらる。一つの仮想ナビゲータが別の仮想ナビゲータをコールする代わりに、仮想ナビゲータは幾つかの仮想ナビゲータの機能を有することが可能である。付加的なオブジェクトクラス(例えば、コンテナ即ち容器)、トランザクション機能及びナビゲータを実現することが可能である。仮想ナビゲータは仮想データ構造以外のリンクされているデータ構造に対する仮想リンクを発生することが可能である。コンポーネントを識別した後にその他の機能を実行することが可能であり、例えば、文獻目録、参考文献、目次、インデックスなどが発生することが可能である。

【図面の簡単な説明】

【図1】 本発明に基づく仮想ナビゲータをサポートするのに適したコンピュータプラットフォームを示した概略ブロック図。

【図2】 電子文書内のコンポーネントの階層を示した概略図。

【図3A】 ベースリンク及び仮想リンクを示した概略図。

【図3B】 ベースリンク及び仮想リンクを示した概略図。

【図4】 仮想ナビゲータが使用される文庫のフローチャート。

【図5】 仮想ナビゲータをカスケード構造とした場合を示した概略図。

【図6】 順番付けリスト駆動ナビゲータのフローチャート。

【符号の説明】

100 コンピュータプラットフォーム

101 電子文書出版システム

102 駆動ナビゲータ

104 デジタルコンピュータ

106 ディスプレイ

108 キーボード

110 マウス

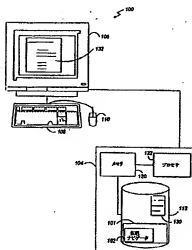
112 大量記憶装置

120 メモリ

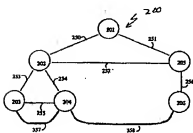
122 プロセサ

130、132 電子文書

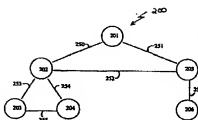
【図1】



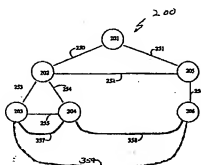
【図3A】



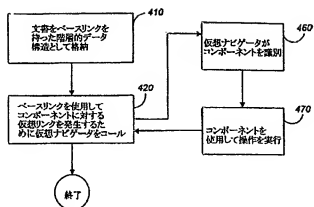
【図2】



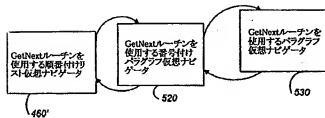
【図3B】



【図4】



【図5】



【図6】

